NETWORK AND METHOD OF CONFIGURING A NETWORK

REFERENCE TO RELATED APPLICATIONS

[0001] This application is a Continuation-in-Part (CIP) of U.S. Non-Provisional Application Ser. No. 10/291,865 entitled "Method And Apparatus for Cluster Interconnection Using Multi-Port Nodes and Multiple Routing Fabrics," filed November 7, 2002 and claims the benefit of U.S. Provisional Application 60/393,936 filed July 2, 2002, which applications are incorporated herein by reference.

BACKGROUND

[0002] In modern computer systems, much of their functionality is realized by the ability to network, that is connect, various computers to provide digital communication. Indeed many interconnection schemes have been developed that meet interconnection needs in various ways. For example, multiprocessor systems can be configured as bus-connected or ring-connected multiprocessor systems. The operation and design constraints of such systems, however, do not lead to designs for reliable and scalable switched networks, especially ones that implement crossbar switches employing wormhole routing. The primary limitation of this type of configuration is that ring topologies are not suitable for wormhole-routed switched networks and result in an unacceptably large hop count between end nodes or endpoints as the network is scaled.

[0003] In another example, the design of bus-oriented interconnection topologies for single-hop communication among multiple transceiver stations is not applicable to scalable switched networks because, among other things, a single-hop interconnection between a large number of nodes is impossible when crossbar switches with a limited number of ports are used. Moreover, such designs use bus-based interconnects which bear little resemblance, if any, to switched interconnects.

[0004] Non-bus-oriented single-hop interconnections are also deficient in a number of ways. For example, such configurations suffer the same limitations as described above while also connecting nodes (or switchless networks) directly. This latter feature limits the applicability of the design to end nodes having a large number of ports and to fabrics having zero switches and hence is inapplicable to the design of switched interconnects.

[0005] In a traditional approach, ServerNet networks have been designed with two ports, also called "colored" ports or "*X*" and "*Y*" ports, connected to two complete, independent groups of crossbar switches. The interconnection group is complete because every end node interfaces with each group of crossbar switches and each group of switches interfaces with every node. Moreover, the interconnection group is independent because ports of one type are only connected to other ports of the same type. For example, each of the *X* ports is only connected via an *X* fabric to other *X* ports and each of the *Y* ports in the network is likewise only connected via a *Y* fabric to other *Y* ports. Note here that an *X* fabric is a group of switches that connect all the *X* ports and only the *X* ports in the network (similarly for *Y* ports). In this way, a fabric of one type is designed independently of other fabrics of other types.

[0006] A particular concern in network design is fault tolerance. With a large scaled system there is insufficient protection against single points of failure because of the large number of components, and it is hard to maintain symmetry because of failed parts. Moreover, scalable topologies (*e.g.* fat trees) offer design points exponentially far apart. In addition, the relative capacity of an end node shrinks as a network grows in size.

[0007] One improved approach has introduced ServerNet Asymmetric Fabrics. With this approach, end nodes are connected using two complete but non-identical groups of switches. Namely, network expansion requires scalable switched networks. The issue, however, is scalable yet highly available fabrics. Hence, there is a further need for optimizing the reliability and performance of scalable switched networks.

[0008] Existing solutions in the area of bus-connected and ring-connected multi-computer systems do not lead to designs for scalable and reliable switched networks because of the operation and design constraints of such solutions. This is especially true in networks configured for use with crossbar switches employing wormhole routing. Moreover, such solutions do not address how a network comprising multiple incomplete fabrics can simultaneously optimize the reliability and the performance of scalable switched networks.

[0009] While the above interconnection schemes provide certain functionality, they are nonetheless limited in at least the ways discussed above. With the advent of network interface cards and other similar devices that provide for multiple ports on one computer system, network design can be expanded beyond the constraints of prior art systems. Importantly, interconnection fabrics need not be constrained to being complete nor colored. Notably, interconnection fabrics should be allowed to be incomplete while allowing for improved fault tolerance and reduced hardware resources. Toward finding an optimal design, however, there exists a need to determine the bounds on various parameters of network designs.

SUMMARY

[0010] An exemplary embodiment may comprise a method for configuring a network. The method comprises assigning a plurality of first nodes as a balanced incomplete block design of the form 2-(v, k, 1) = b, wherein v first nodes, arranged in b groups of k first nodes, are interconnected such that a pair of first nodes appears in only one group of the b groups. The method also comprises assigning a plurality of sets of second nodes wherein each first node is associated with at least one set of second nodes, and determining network paths from each second node of the plurality of sets of second nodes to every other second node.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The accompanying drawings, which are incorporated in and form a part of this specification, illustrate exemplary embodiments and, together with the description, serve to explain the principles of the present disclosure.

[0012] Figure 1 is a network diagram according to an exemplary embodiment for interconnecting seven elements each with three ports.

[0013] Figure 2 is a network diagram according to an exemplary embodiment for interconnecting three elements using three fabrics.

[0014] Figure 3 is a network diagram according to an exemplary embodiment for interconnecting four elements using six fabrics.

[0015] Figure 4 is a network diagram according to an exemplary embodiment for interconnecting five elements using ten fabrics.

[0016] Figure 5 is a network diagram according to an exemplary embodiment for connecting 65 nodes using five elements and ten fabrics.

[0017] Figure 6 is a block diagram according to an exemplary embodiment of a five-element network comprising two fabrics.

[0018] Figure 7 is a block diagram according to an exemplary embodiment of a partial five-element network comprising an X fabric.

[0019] Figure 8 is a block diagram according to an exemplary embodiment of a partial five-element network comprising a Y fabric.

[0020] Figure 9 is a block diagram according to an exemplary embodiment of various endpoints connected to a node through X switches.

[0021] Figure 10 is a block diagram according to an exemplary embodiment of various endpoints connected to a node through Y switches.

[0022] Figure 11 is a block diagram according to an exemplary embodiment of various dual-ported endpoints connected to a node connected through a collection of *X* and *Y* switches.

[0023] Figure 12 is a block diagram according to an exemplary embodiment of various endpoints and nodes connected as an *X* fabric.

[0024] Figure 13 is a block diagram according to an exemplary embodiment of various endpoints and nodes connected as a *Y* fabric.

[0025] Figure 14 is a block diagram according to an exemplary embodiment of various endpoints and nodes connected as a collection of an *X* and *Y* fabric.

[0026] Figure 15 is a block diagram according to an exemplary embodiment of a nine-element network comprising two fabrics.

[0027] Figure 16 is a block diagram according to an exemplary embodiment of a partial nine-element network comprising an *X* fabric.

[0028] Figure 17 is a block diagram according to an exemplary embodiment of a partial nine-element network comprising a *Y* fabric.

[0029] Figure 18 is a block diagram according to an exemplary embodiment of various endpoints connected to a node through *X* switches.

[0030] Figure 19 is a block diagram according to an exemplary embodiment of various endpoints connected to a node through *Y* switches.

[0031] Figure 20 is a block diagram according to an exemplary embodiment of various endpoints connected to a node through a collection of *X* and *Y* switches.

[0032] Figure 21 is a block diagram according to an exemplary embodiment of various endpoints and nodes connected as an *X* fabric.

[0033] Figure 22 is a block diagram according to an exemplary embodiment of various endpoints and nodes connected as a *Y* fabric.

[0034] Figure 23 is a block diagram according to an exemplary embodiment of a 9-node network.

[0035] Figure 24 is a block diagram according to an exemplary embodiment of various endpoints connected as a fabric.

[0036] Figure 25 is a block diagram according to an exemplary embodiment of various endpoints connected as a fault-tolerant fabric.

[0037] Figure 26 is a block diagram of an exemplary computer system.

DETAILED DESCRIPTION

[0038] The drawing and description, in general, disclose a network and a method of configuring a network using a multi-fabric design process. This multi-fabric design process greatly facilitates the design of networks of various topologies and results in networks that are advantageous for a variety of reasons, as will be discussed below. For example, multi-fabric design enables the designer to find an optimal design in which each class of items appears in only the desired number of fabrics, in other words, without over-designing the network. Redundant paths may be provided in the network if desired by mapping, for example, two logical fabrics in the mathematical design into one physical fabric. Multi-fabric design may be used to design networks having symmetric or asymmetric fabrics, crossbar-only interconnects (single-hop networks), etc. An exemplary embodiment of the multi-fabric design process to be disclosed herein may be summarized in the following four steps.

[0039] Step 1. The starting point is a combinatorial design, generally a BIBD (Balanced Incomplete Block Design) -- 2-(v,b,r,k,λ) -- where small values of r are preferred. (v items are grouped into b blocks of size k such that k < v and each item is in exactly r blocks and each set of 2 items, i.e. each pair, appears together in at least λ groups, as will be described below.)

[0040] Step 2. (optional) Partitioning the logical design of Step 1, if it is a partitionable BIBD. Graph-theoretic techniques are used when b=2; combinatorial techniques, when b>2.

[0041] Step 3. Each mathematical "item" from the previous steps is mapped into a "class." A class may either be a singleton computer node or may have internal structure. If latter, the "class switches" may be shared between the different fabrics that the class connects into. Classes may also be assembled from disjoint subclasses, interconnectivity between which is deferred until Step 4. Recursive application of MFD is optional.

[0042] Step 4. The "blocks" from Steps 1 and 2 -- a.k.a. logical fabrics -- are mapped into physical fabrics. Since k < v, each fabric is partial, in that not all the nodes of the topology are reachable through it. A fabric may be as simple as either a single link between a pair of classes or a singleton switch that connects all of the links that need to be connected. Generally, it is a network, possibly designed through recursive application of MFD.

[0043] If class sharing is used in Step 3, then the resulting topology will have fewer physical fabrics than logical ones. When there are only two physical fabrics but b>2, the special case of asymmetric fabrics occurs. Otherwise, when classes are implemented using singleton nodes in Step 3, and when singleton crossbar switches are used to realize physical fabrics in Step 4, the special case of crossbar-only interconnects (COIs) occurs. COI topologies uniquely extend the size of the largest system in which every pair of nodes is interconnected via a single crossbar switch.

[0044] It has thus been found that network designs with various advantages can be formed from mathematical concepts of balanced incomplete block designs (BIBDs). From these BIBDs a logical or virtual mapping can be derived for a network from which, in turn, a physical design is derived. In order to understand the present disclosure, however, it is useful to understand combinatorial block design and, in particular, balanced incomplete block design (BIBD). A block is a subset, *s*, of a set of elements, *S*, where block design considers choosing blocks with certain properties. A block design is called incomplete if at least one block does not contain the entire set of elements. A block design is balanced if each block has the same number of elements and each pair of elements occurs in a block the same number of times. For the purposes of the present approach, BIBD theory is used to design networks that have predetermined characteristics or properties.

[0045] With a BIBD, a pair (V, B) exists where V is a set of v elements and B is a collection of b blocks that are subsets of k elements of V such that each element of V is contained in exactly r blocks and any two-subsets of V is contained in exactly λ blocks. The variables v, b, r, k, and λ are parameters of a BIBD family also referred to as 2-(v, b, r, k, λ) block design. In such a design, b groups are needed to connect v elements arranged in groups of k, such that each pair of elements appears in exactly λ groups. Two conditions are established for the existence of a BIBD: (i) $r(k\text{-}1) = \lambda(v\text{-}1)$, and (ii) $vr = bk$. A consequence of these conditions is that three parameters, v, k, and λ, determine the remaining two parameters, r and b, from equations i and ii as follows:

$$r = \frac{\lambda(v-1)}{k-1} \qquad (1), \text{ and}$$

$$b = \frac{vr}{k} . \qquad (2)$$

[0046] With regard to equation 1, consider that an element, x, occurs in r blocks. Further consider that in each of those blocks, x is paired with k-1 other elements. Thus, x occurs in $r(k\text{-}1)$ pairs of co-occurring elements. Further note that x must be paired with all other v-1 elements exactly λ times (*i.e.*, $\lambda(v\text{-}1)$) and equation 1 is therefore proven. It is straightforward to see that each block, b, contains k elements for a total of bk elements. Also, each element occurs in r blocks and since there are v elements the total is vr, thus we have equation 2.

[0047] Accordingly, a BIBD (v, b, r, k, λ) design can also be referred to as a (v, k, λ) design. The notation 2-(v, k, λ) = b is also used, since BIBDs are t-designs of the form t-(v, k, λ) with

$t=2$. Note that when $\lambda=1$ (*i.e.*, 2-(v, k, 1)), the notation $S(2, k, v)$ is also used denoting that these are Steiner systems (named after nineteenth century geometer Jakob Steiner). With regard to Steiner systems, given three integers, t, k, v, such that $2 \leq t < k < v$, a Steiner system $S(t, k, v)$ is a set V of v elements together with a family, B, of subsets of k elements of V (*i.e.*, blocks) with the property that every subset of t elements of S is contained in exactly one block. Recall that in BIBD, $t=2$. These systems therefore determine the number of groups that are needed to connect v elements, arranged in groups of k, such that a pair (*i.e.*, "2-") appears in exactly λ groups, where in a Steiner system $\lambda=1$ group.

[0048] Moreover, from Fisher's inequality, $b \geq v$. Designs with $b=v$ and $r=k$ are called symmetric designs where every block contains k elements and every element occurs in r blocks. Also, every pair of elements occurs in λ blocks, and every pair of blocks intersects in λ elements.

[0049] Whereas BIBD designs can be quite complicated they can be represented in a two-dimensional, k x b array in which each column contains the elements forming a block. For example, consider the 2-(9, 3, 1) = 12 design:

$$\text{Elements}\left\{\overbrace{\begin{array}{cccccccccccc} 0&0&0&0&1&1&1&2&2&2&3&6\\ 1&3&4&5&3&4&5&3&4&5&4&7\\ 2&6&8&7&8&7&6&7&6&8&5&8 \end{array}}^{\text{Blocks}}\right.$$

[0050] Here, for example, the first column represents the block containing elements e_0, e_1, and e_2 and the twelfth column represents a block having elements e_6, e_7, and e_8. In a larger design, letters can be used to represent blocks with more than 10 elements. The sequence 0, 1, …, 9, a, b, …, z can represent designs with up to 36 elements (*i.e.*, 10 numerically represented elements and 26 alphabetically represented elements). Thus, the following 2-(16, 4, 1)=20 design can be represented as follows:

$$\text{Elements}\left\{\overbrace{\begin{array}{cccccccccccccccccccc} 0&0&0&0&0&1&1&1&1&2&2&2&2&3&3&3&3&4&5&6\\ 1&4&7&a&d&4&5&6&9&4&5&6&8&4&5&6&7&8&9&7\\ 2&5&8&b&e&7&b&8&c&c&7&9&a&9&8&a&b&b&a&c\\ 3&6&9&c&f&a&d&e&f&e&f&b&d&d&c&f&e&f&e&d \end{array}}^{\text{Blocks}}\right.$$

With a design in hand, a BIBD can be further described by an incidence matrix A which has the blocks as its columns and elements (*e.g.*, nodes) as the rows. Thus, an entry, $a_{i,j}$ of the incidence matrix A is equal to one if the ith element resides in the jth block, otherwise it is equal to zero. For example, for a symmetric design with N elements, the incidence matrix is an NxN matrix.

Accordingly, the 2-(9, 3, 1) = 12 design

$$\text{Elements}\left\{\overbrace{\begin{array}{cccccccccccc}\hline 0&0&0&0&1&1&1&2&2&2&3&6\\ 1&3&4&5&3&4&5&3&4&5&4&7\\ 2&6&8&7&8&7&6&7&6&8&5&8\\ \hline\end{array}}^{\text{Blocks}}\right.$$

described above is represented by the following incidence matrix:

$$A_{2\text{-}(9,3,1)} = \left.\overbrace{\begin{bmatrix} 1&1&1&1&0&0&0&0&0&0&0&0\\ 1&0&0&0&1&1&1&0&0&0&0&0\\ 1&0&0&0&0&0&0&1&1&1&0&0\\ 0&1&0&0&1&0&0&1&0&0&1&0\\ 0&0&1&0&0&1&0&0&1&0&1&0\\ 0&0&0&1&0&0&1&0&0&1&1&0\\ 0&1&0&0&0&0&1&0&1&0&0&1\\ 0&0&0&1&0&1&0&1&0&0&0&1\\ 0&0&1&0&1&0&0&0&0&1&0&1 \end{bmatrix}}^{\text{Blocks}}\right\}\text{Incidence of Elements}$$

[0051] From a BIBD, network designs can in turn be generated by identifying certain correspondences. For example, given the blocks of a BIBD 2-(v, k, λ), the mapping between BIBD and network design is given by the following table.

Table 1: Mapping from Block Design to Network Design

Block Design	**Network Design**
Elements	Nodes or classes of nodes
Blocks	Fabrics or interconnections
λ (*i.e.*, the number of pair-wise occurrences of elements)	The number of fabrics that interconnect a pair of nodes or classes of nodes
r (*i.e.*, total occurrence of an element)	Degree of a node or the out-degree of a class
k (*i.e.*, block size)	Length of routing

A solution to a BIBD provides a partition of the *v* elements into subsets such that there are exactly λ subsets for each pair of the elements and the distance between any two elements is at most *k*-1 and, at best, $\lceil \log_m k \rceil$ where the operator $\lceil \bullet \rceil$ denotes rounding up to the nearest integer, and the radix, *m*, is a technology-dependent constant.

[0052] Thus, the two important parameters of a block design are *k* and *r*. The size *k* of each block determines the maximum length of routing, and the total number of occurrences, *r,* of each element determines the degree requirement for such element in the target network. Particularly, smaller *k* leads to a better bound on the length of routing and smaller *r* requires a smaller number of network interface ports at endpoints in the target network.

[0053] For λ=1 (*i.e.*, a Steiner system), each block of size *k* is unique for all possible pairs of *k* elements that it contains. That implies that each possible pairing of elements in a block corresponds to a unique candidate edge for the target topology. Furthermore, since such an edge never occurs in any other block, the virtual rings corresponding to the blocks are mutually edge-disjoint. Thus, each block of size *k* can induce a complete graph of *k* elements. In graph theory, any graph with *k* elements can be embedded into a complete graph with *k* elements.

[0054] Using the foregoing principles, a class of interconnect networks and multiple incomplete fabric interconnect systems are disclosed that can be used to simultaneously scale the performance and the reliability of either multi-computer cluster systems, switched input/output systems or switched processor-memory systems, while using fewer components than a traditional approach. In doing so, each end node, such as a computer, network-attached I/O device, or processor, has more than two network interface ports. The multiple ports can be provided either through the use of computers with network interface cards (NICs), each having one or more ports, or through the use of multi-port I/O nodes, or through the use of switched processor-memory chipsets. Preferably, this approach takes advantage of the dual-ported and multi-ported NICs that are a key part of widely used networks including, for example, ServerNet networks designed by the Hewlett-Packard Corporation. Such an approach can also be implemented in networks including Ethernet, GigaNet, Fibre Channel, ATM (Asynchronous Transfer Mode), RDMA-enabled Ethernet, PCI Xpress, InfiniBand, multiwavelength optical networks or other switched networks that have either been developed or will be developed in the future. Switched processor-memory subsystems include, but are not limited to, Sun UE10K, SGI Origin, Intel Profusion Chipset, and Compaq Alpha EV7. Switched IO subsystems include, but are not limited to, ServerNet, PCI Xpress, Stargen, InfiniBand and Rapid I/O.

[0055] In regards to the present approach, consider that a fabric is a collection of routers, switches, forwarding nodes, and links that interconnect a set of nodes. In the present discussion reference will be made to routers, switches, forwarding nodes, and other types of switching or interconnection devices that provide a path for and relay data between end nodes, in that they forward data from a receiving port to a sending port; it should be noted, however, that where a specific device is mentioned, the broader applicability of the present disclosure is intended to be illustrated with such particular example.

[0056] Further consider that a node may have one or more NICs (network interface cards), each with two or more ports. Among other things, each port allows a node to be on a distinct fabric. In one embodiment, fabrics, ports, and routers have color restrictions. For example, ports and routers are either red or green (note that the coloring described here can also be described with reference to X and Y designations). In a coloring scenario, it is illegal to connect a red port or router to a green port or router; *i.e.*, there is either a red fabric or a green fabric. Stated another way, each fabric connects either red ports using only red routers (*i.e.*, a red fabric) or, alternatively, green ports using only green routers (*i.e.*, a green fabric), but there is no interconnection between colors. The problems underlying network topology design are minimizing diameter, maximizing bisection width, minimizing the number of routers, avoiding excessive link contention and avoiding hot links, and these problems are assumed to be important here. In some embodiments, however, coloring constraints are eliminated.

[0057] Several issues unique to multi-fabric topologies will now be examined. More particularly, a determination of how large each fabric needs to be will be examined. As a fundamental matter, fabrics collectively provide at least one path between each pair of nodes. While this can be accomplished with a large number of fabrics, a number of fabrics larger than necessary can waste routers by making redundant connections between nodes, thereby increasing costs.

[0058] A determination of how many fabrics are needed is also important: This is an important yet difficult matter to determine. In one embodiment, the number of fabrics is bounded either above or below, or both above and below, to determine an approximation for the optimal solution. As before, this will ensure that each pair of nodes appears together in at least one fabric, given a specific fabric size.

[0059] It is evident that redundant connections are inevitable and indeed desirable in all but the simplest of cases. Should redundant connections be present, a pair of nodes will co-occur in more than one fabric. Further, within each fabric, distance between nodes may vary from pair to pair. Rather than have some pair of nodes be far apart in all fabrics — and have other pairs be close together in more than one fabric — the multiple fabrics may be so arranged that each additional fabric causes the shortest available distances between some formerly far nodes to become smaller, perhaps at the expense of the additional fabric's distance between already closely connected nodes.

[0060] It should be noted that the multi-fabric design problem discussed here is different from the problem of multiple ports in one fabric. For example, multiple fabrics according to the present approach are likely to provide better protection for nodes against faults and congestion. Moreover, the diameter of a multi-fabric network is generally smaller than that of its single-fabric counterpart. This not only reduces the number of outstanding packets necessary for keeping pipelines full but also lowers the impact of output-port contention on link utilization. In effect, the multiple fabrics create congestion-containment domains or routing domains.

[0061] With the understanding that multi-fabric designs provide advantages over traditional solutions, we now turn to implementations of multi-fabric designs. Although several embodiments will be described, it can be understood that the present disclosure is not limited to the described embodiments.

[0062] Consider the following problem: given n nodes where each node connects to p different fabrics, what is (1) the minimum number of fabrics and (2) the minimum fabric cardinality (the number of nodes in a fabric) required to ensure full connectivity between all nodes? Furthermore, what is a minimal assignment of connections to fabrics?

[0063] While the present discussion applies to both colored and non-colored fabric implementations, those implementations that completely ignore color will be considered first. In doing so, it has been found that n nodes can be connected using k fabrics of cardinality m such that

$$\left\lceil \frac{\binom{n}{2}}{\binom{m}{2}} \right\rceil \le k \quad \text{(Equation 3)}$$

where $\lceil \bullet \rceil$ represents rounding up to the next whole number and $\binom{i}{j}$ represents the binomial coefficient C_j^i such that

$$C_j^i = \frac{i!}{j!(i-j)!} \underline{\Delta} \binom{i}{j}$$

denotes the number of different sub-populations of size j that can be chosen from a set of size i (*i.e.*, i choose j). The above inequality follows from the requirement that every pair of nodes must be connected by at least one fabric. Moreover, since each fabric generates at most $\binom{m}{2}$ pairs, and full connectivity among the nodes requires at least $\binom{n}{2}$ pairs, the resulting lower bound on the number of fabrics, k, follows.

[0064] In considering the lower bound on fabric size, the neighborhood relationships of a single node are examined to impose a constraint that a node has to connect to all of its peers through a finite (and preferably small) number of ports. Using concepts from graph theory, consider that a node forms a vertex on a graph and an edge is an unordered pair of distinct vertices. It has therefore been found that with n nodes, each having p ports that are connected using fabrics of vertex cardinality m (*i.e.*, the number of vertices),

$$m \ge \left\lceil \left(\frac{n+p-1}{p} \right) \right\rceil \quad \text{(Equation 4).}$$

[0065] Notably, because a node has only p ports, it cannot connect to more than p fabrics. Moreover, because each fabric offers connections to only m-1 other neighbors, m must be large enough to cover all neighbors. Therefore,

$$\left\lceil \frac{n-1}{m-1} \right\rceil \le p$$

such that

$$(n-1) \le p(m-1)$$
$$(n-1) \le pm - p$$

which we manipulate into the form

$$m \ge \left\lceil \left(\frac{n+p-1}{p} \right) \right\rceil .$$

[0066] A straightforward example reinforces the above principles. As shown in Figure 1, consider interconnecting seven nodes 10, corresponding to the previously described elements, each with three ports (e.g., 12). In fact, this problem corresponds to the BIBD of 2-(7, 3, 1) = 7, that is, seven groups are needed to connect seven elements, arranged in groups of three, such that each pair of elements appears in exactly one group. Since each node must communicate to its 6 peers via only 3 ports, each fabric must have a size (*i.e.*, vertex cardinality) of at least 3, according to Equation 4:

$$m \ge \left\lceil \frac{7+3-1}{3} \right\rceil$$

$$m \ge 3.$$

Moreover, the minimum number of fabrics, according to Equation 3 is

$$\left\lceil \frac{\binom{7}{2}}{\binom{3}{2}} \right\rceil \le k$$

$$7 \le k.$$

[0067] In this example, it is important to note that these lower bounds provide tight bounds. Indeed, the fact that both these lower bounds are tight, at least for certain cases, is illustrated by an assignment of nodes to fabrics as shown in Table 2.

Table 2: Assignment of Nodes

Fabric	1st Node	2nd Node	3rd Node
1	Node 1	Node 2	Node 3
2	Node 1	Node 4	Node 5
3	Node 1	Node 6	Node 7
4	Node 2	Node 4	Node 6
5	Node 2	Node 5	Node 7
6	Node 3	Node 5	Node 6
7	Node 3	Node 4	Node 7

[0068] This shows that seven fabrics 14 of size three are not merely the minimum requirement but are also sufficient in this case. The topology of these interconnection fabrics is further shown in Figure 1.

[0069] It is found that the coloring of fabrics adds strong constraints to the fabric partitioning problem. In fact, multi-fabric design with nodes having only two ports, where each port has a different color, may be impractical in all but the most trivial cases. Consider that if each node has only two ports, one red and one green, then at least one fabric must connect all of the nodes. This result can be shown by contradiction as follows. For example, suppose to the contrary that a node *n* connects to a red fabric F_R and a green fabric F_G in such a fashion that neither F_R nor F_G connects all of the nodes together, that is,

$F_R \subset N$ and

$F_G \subset N$

where *N* is the set of all nodes. Thus, either

$$F_R \cup F_G = N$$

or

$$F_R \cup F_G$$

is strictly a proper subset of N.

[0070] Since the latter case would imply incomplete connectivity for N, only the former can be accepted. Therefore,

$$F_R \cup F_G = N .$$

Since node *n* belongs to both red and green fabrics, there must exist nodes

$$n_R \in F_R$$

and

$$n_G \in F_G$$

such that

$n_R \neq n_G$,

$n_R \notin F_G$, and

$n_G \notin F_R$.

[0071] In order to achieve complete connectivity between all pairs of nodes, it is therefore necessary to add a fabric, say F_X, that will connect n_R to n_G where F_X could be neither red nor green. Because it is impossible to connect n_R to n_G using colored fabrics as constrained above, a contradiction exists. The only available ports for connecting to F_X, however, are green on n_R

and red on n_G. Because our supposition has been contradicted, the opposite must be true, that is, at least one fabric must connect all the nodes.

[0072] It is because of this result that multi-fabric design was not attempted in traditional systems with only two ports, such as ServerNet I. With the availability of multi-port equipment, such as dual-PCI Compaq Professional Workstation platforms that support two NICs, each with two ports, called the *X* and *Y* ports, multi-fabric designs became feasible and, indeed, desirable because of their advantages. In implementing multi-fabric designs, it has been found, for example, that ServerNet II offers a flexible coloring of ports so that even with only one ServerNet II NIC, a node can have two ports of the same color. Partitioned fabric designs are therefore practical even in systems having only one ServerNet II NIC per node, but not practical in systems with only one ServerNet I NIC per node.

[0073] The further advantage of ServerNet II's flexible coloring of NIC ports becomes apparent when the fabric-partitioning solution described in Table 2 is examined. If all ports were the same color, the solution described above would function properly because fabric coloring would not be an issue. For nodes with a pair of ServerNet I NICs, however, two of the four ports on each node would be *X* ports and the other two would be *Y* ports. ServerNet I NICs and routers set and check the path bit, identifying a path as either *X* or *Y*, in almost all packets (except for default ports on routers); and, in general, it is not possible to route packets between *X* and *Y* ports and/or routers. With regard to Table 2, rows of the table (or, fabrics) should be colored in such a way that no node appears in more than two fabrics of the same color.

[0074] Let us now consider a specific impossibility argument in the context of Table 2 and then a general theorem for partitions with an odd number of fabrics. Without loss of generality, suppose that a fabric, say Fabric One 16, is colored red. Since Node One 20 has only two red ports and it appears on a total of three fabrics (Fabric One 16, Fabric Two 22 and Fabric Three 24), it must be that at least one of the other two fabrics 22 and 24 on which it appears must be green. Again, without loss of generality, suppose that a second fabric, say Fabric Two 22, is colored green. Applying the same argument to Node Two 26, either Fabric Four 30 or Fabric Five 32 must be green. Suppose that Fabric Four 30 is green. Next, consider Fabric Seven 34. Since both the green ports on Node Four 36 are used up, this fabric 34 must be colored red. Doing so uses up both the red ports on Node Three 40. Hence, Fabric Six 42 must be colored green. Doing so uses up both the green ports at Node Five 44. Hence, Fabric Five 32 must be colored red. Now, we need to assign a color to Fabric Three 24 which connects Nodes One 20, Six 46 , and Seven 50, but both green ports are used up on Node Six 46 as well as both red ports on Node Seven 50. It is therefore impossible to pick a color for Fabric 3 24.

[0075] In proceeding, we will further be constrained by the mathematical impossibility of coloring an odd number of fabrics with two colors — say, red and green — if each node has an equal number of red and green ports.

[0076] Having now considered lower bounds, it is important to consider also upper bounds. Although redundancy may be inevitable, redundancy can be quantified by fixing at the outset the number of nodes that will co-occur in all fabrics. Optimal solutions may not always be possible, but an interesting effect is that we can always come up with a feasible solution. Since the solutions so found yield closed-form expressions for both the size and the number of fabrics, those expressions serve as upper bounds on the respective quantities. The key observation here is that many nodes may connect to the same collection of fabrics, and these equivalent nodes can be handled together in an equivalence class. Equivalence classes can be thought of as nodes that always co-occur in fabrics. Equivalence classes are a natural algebraic abstraction for the multi-fabric design problem because connectedness, the primary relationship of interest here, is, algebraically speaking, an equivalence relation in that it is trivially reflexive, symmetric and transitive. A solution is constructed by increasing the number of equivalence classes.

[0077] For illustrative purposes only, we first consider a restricted set of embodiments of the present teachings where each fabric interconnects exactly two equivalence classes and where each class is a simple grouping of unconnected singleton endpoints. While arbitrary, this restriction allows us to demonstrate the present teachings using graphical techniques as follows. In the graphs of Figures 2, 3, and 4, each vertex 60, 62 and 64 represents an equivalence class, and each edge 66, 70 and 72 represents the fabric that interconnects the two classes corresponding to its two vertices. For the degenerate and trivial cases of one or two classes (not shown graphically), a single fabric connects all of the nodes, and each node needs only one fabric connection. That stated, we turn to more useful designs.

[0078] In partitioning nodes into three equivalence classes, S_1 60, S_2 62, and S_3 64, as shown in Figure 2, each class connects to two fabrics and there are three total fabrics 66, 70 and 72. Fabric F_{12} 66 connects all of the nodes in classes S_1 60 and S_2 62, Fabric F_{13} 70 connects all of the nodes in classes S_1 60 and S_3 64, and fabric F_{23} 72 connects all of the nodes in classes S_2 62 and S_3 64. With four equivalence classes 80, 82, 84 and 86, as shown in Figure 3, each class (e.g., 80) connects to three fabrics (e.g., 90, 92 and 94) and there are $\binom{4}{2} = 6$ different fabrics 90, 92, 94, 96, 100, and 102 in all. With five equivalence classes 110, 112, 114, 116 and 120, as shown in Figure 4, each class (e.g., 110) connects to four fabrics (e.g., 122, 124, 126 and 130) and there are $\binom{5}{2} = 10$ total fabrics 122, 124, 126, 130, 132, 134, 136, 140, 142 and 144.

[0079] More particularly, the graph of Figure 4 represents a 64-node cluster where each class (e.g., 110) has four connections (e.g., 122, 124, 126 and 130). In an embodiment this is achieved with nodes having two ServerNet NICs, each with an *X* port and a *Y* port. With these specifications, the network of Figure 5 is built. In order to simplify the design, the nodes are partitioned into equivalence classes where each fabric is a pairing of equivalence classes. With five equivalence classes, *S*1-*S*5 150, 152, 154, 156 and 160 as shown in Figure 5, each node (e.g., 150) connects to four fabrics (e.g., 162, 164, 166 and 170) and there are ten total fabrics 162, 164, 166, 170, 172, 174, 176, 180, 182 and 184. Rounding the number of nodes up to 65, we have 13 (*i.e.*, 65/5=13) nodes per class with each fabric connecting 26 (*i.e.*, 2x13=26) nodes. Note that if each fabric were a simple Steiner tree, 26 nodes would require 6 6-port routers such that the 64-node configuration can be done in 6*10=60 routers. The complete solution is therefore shown in Figure 5. Coloring constraints are easily satisfied because the perimeter of the pentagon can be built with *X* fabrics 162, 170, 174, 182 and 184 (shown as solid lines) and the core can be built with *Y* fabrics 164, 166, 172, 176 and 180 (shown as dashed lines). Indeed, an important result is that it provides for fault-tolerant systems; the occurrence of a failure anywhere in the system will not render the rest of the system useless. Moreover, the present approach provides for redundant interconnection paths such that if a failure does occur, a redundant path is available.

[0080] Indeed, the above technique of "fabrics as class pairs" can be extended to more general network configurations with the understanding of equivalence classes. For interconnecting nodes with *p* ports, there are (*p*+1) equivalence classes. With *n* such nodes, the vertex cardinality of each fabric is given by

$$m = 2\left\lceil \frac{n}{p+1} \right\rceil \qquad \text{(Equation 5).}$$

[0081] Notably, the concept of equivalence classes plays an important role in this solution as will be further explained. Bisection bandwidth (the minimum number of paths, when considering all possible partitions, which must cross if a design is partitioned into two equal halves) is observed to be good for the resulting network topologies, but it can be difficult to compute because the number of classes is usually odd. Using a tree for each fabric, the 64-node topology discussed above has a bisection bandwidth of greater than ten (10) links. Because of the high cost of the 60 routers, adoption of such a design can be difficult. The quality of solutions generated — as quantified by, say, bisection width and number of routers needed — depends upon the size of equivalence classes. The smaller the class size, the smaller is the number of connections that repeat in all fabrics. Because each node participates in p fabrics, the connections within a node's equivalence class are redundantly repeated (p-1) times. Thus, it can be seen that the larger the class size, the greater is the waste. When each fabric connects only two simple classes (which is not a requirement of the present teachings but rather an arbitrary restriction needed for illustrative purposes and only in these first few embodiments), given the lower bounds on fabric size discussed above, class size must be at least

$$\text{Class size} \geq \left\lceil \frac{n+p-1}{2p} \right\rceil \qquad \text{(Equation 6).}$$

[0082] For the 64-node topology shown in Figure 5, the bounds of Equations 3 and 4 suggest a minimum fabric cardinality of 17, and a minimum fabric count of 15. At 26, the network of Figure 5 has sufficient fabric cardinality, but, subjectively, a larger than necessary number of fabrics may be in use. For fabric cardinality 26, the lower bound on the number of fabrics is 7, according to Equation 3. At 10, the number of fabrics in the network of Figure 5 is significantly above that minimum value. Whereas the illustrative discussion above confirms the feasibility of designing networks with multiple fabrics, it also shows that the illustrative "fabrics as class pairs" approach does not always yield either optimal or near-optimal fabric count for a given fabric cardinality. The discussion of this first set of embodiments is nevertheless useful because it does yield tight lower bounds on fabric size and fabric cardinality, as well as provides upper bounds through the construction of multi-fabric designs in which each fabric interconnects a pair of equivalence classes.

[0083] Furthermore, certain designs produced using the "fabrics as class pairs" approach are guaranteed to satisfy the hard-to-satisfy color constraint of multi-fabric partitioning described earlier. The bounding results of the "fabrics as class pairs" design are therefore summarized here for the optimal fabric size

Fabric Parameter	**Lower Bound**	**Upper Bound**
Optimal Fabric Size (m_O)	$\left\lceil \frac{n+p-1}{p} \right\rceil$	$2\left\lceil \frac{n}{p+1} \right\rceil$

and the optimal number of fabrics.

Fabric Parameter	Lower Bound	Upper Bound
Optimal Number of Fabrics (k_O)	$\left\lceil \frac{\binom{n}{2}}{\binom{m}{2}} \right\rceil$	$\binom{p+1}{2}$

[0084] The discussion above has demonstrated that whereas the lower bounds are tight, the upper bounds are not. The discussion that follows describes embodiments that, instead of starting with arbitrary groupings, translate BIBDs into network designs in order to equal or more closely approximate the lower bounds on fabric count and cardinality.

[0085] In a first example a 2-(5, 2, 1) = 10 BIBD will be described. Recall that this BIBD was discussed with reference to Figure 4 as a BIBD where 10 groups are needed to connect five elements 110, 112, 114, 116 and 120, arranged in groups of two, such that each pair of elements appear in one group. Figure 4 is redrawn as Figure 6 for clarity in the discussion to follow. This BIBD therefore corresponds to a design of 5 elements 190, 192, 194, 196 and 200, with 2 elements per group, resulting in 10 groups. Where the 5 elements are nodes *V*1-*V*5 190, 192, 194, 196 and 200, the 10 groups are therefore the node-to-node connections F12 202 (otherwise identified as $\{V1, V2\}$ in the fabric equation below), F23 204, F34 206, F45 210, F15 212, F13 214, F14 216, F25 220, F24 222, and F35 224:

$$F \rightarrow \begin{Bmatrix} \{V1,V2\},\{V1,V3\},\{V1,V4\},\{V1,V5\},\{V2,V3\}, \\ \{V2,V4\},\{V2,V5\},\{V3,V4\},\{V3,V5\},\{V4,V5\} \end{Bmatrix}$$

[0086] As shown in Figure 6, this collection of groups 204-224 can therefore be considered a fabric, *F*. Notably, the logical groups 204-224 comprising fabric *F* can be partitioned across two partial fabrics to be called *X* fabric, F_X, consisting of node-to-node connections F12 202, F23 204, F34 206, F45 210 and F15 212, and *Y* fabric, F_Y, consisting of node-to-node connections F13 214, F14 216, F25 220, F24 222, and F35 224. In particular we note the following partitioning of the fabric, *F*:

$$F_X \rightarrow \{\{V1,V2\},\{V2,V3\},\{V3,V4\},\{V4,V5\},\{V1,V5\}\}$$

$$F_Y \rightarrow \{\{V1,V3\},\{V3,V5\},\{V2,V5\},\{V2,V4\},\{V1,V4\}\}$$

Thus, the union of F_X and F_Y provides for the fabric $F=F_X \cup F_Y$.

[0087] The outer ring of groups serially connects nodes *V*1 190 to *V*2 192 to *V*3 194 to *V*4 196 to *V*5 200 and back to *V*1 190 (as shown in Figure 6). The outer ring is, in one instance, an *X* fabric, F_X, as shown in Figure 7. An inner star of groups serially connects nodes *V*1 190 to *V*3 194 to *V*5 200 to *V*2 192 to *V*4 196 and back to *V*1 190. This star pattern can be reorganized in the form of a ring called a *Y* fabric, F_Y, with identical connections as shown in Figure 8. Accordingly, the collection of groups, *F*, as shown in Figure 6 can be redrawn as the union of two rings of groups, F_X and F_Y (*i.e.*, $F=F_X \cup F_Y$) as shown in Figures 7 and 8, respectively.

[0088] We now turn to what has been referenced above as classes or equivalence classes of nodes. An equivalence class is a group of similarly connected nodes or endpoints. For clarity of discussion, we will call these "endpoints" while using the term "node" for the various nodes *V*1-*V*5. In the field to which it pertains, either term, "endpoint" or "node," or even other terms (*e.g.*, "port"), may be used to describe the same items. Accordingly, no definitions are made here. Rather, the usage of specific terms is meant to lend toward understanding the present disclosure.

[0089] Unlike the simple equivalence classes of Figure 5, which consisted only of unconnected singleton endpoints, the equivalence classes of Figures 6 to 8 are internally connected as shown in Figures 9 to 11. There are two principal advantages of such internal connectivity within a class. First, the number of physical network interface ports at endpoints does not need to precisely match the rank (in the BIBD) of the class containing that endpoint. Second, the switches and routers used for internal connectivity within a class can be shared between certain groups of classes. In particular, it is possible and, in accordance with the principles of the current teachings, advantageous that class routers be shared between those groups that have a non-empty intersection and, after partitioning, still map into the same fabric, as described below.

[0090] As shown in Figure 9, four endpoints, N_1 230, N_2 232, N_3 234, and N_4 236, are connected to a six-port switch 240 configured as a 4-in-2-out switch C_X^{1-4} (here *X* denotes the "*X*" fabric). Similarly, four endpoints, N_5 242, N_6 244, N_7 246, and N_8 250, are connected to a 4-in-2-out switch C_X^{5-8} 252; and four endpoints, N_9 254, N_{10} 256, N_{11} 260, and N_{12} 262, are connected to a 4-in-2-out switch C_X^{9-12} 264. Here, the collection of 12 nodes 230-236, 242-250 and 254-262 can be considered as an equivalence class of nodes (endpoints) connected to a node (for example, node *V*1 190 of Figure 7). The following notation, therefore describes the above connections into the *X* fabric for node *V*1 190:

$$V1_X \rightarrow \{C_X^{1-4}(N_1,N_2,N_3,N_4), C_X^{5-8}(N_5,N_6,N_7,N_8), C_X^{9-12}(N_9,N_{10},N_{11},N_{12})\}$$

[0091] Each collection of four nodes mentioned above is considered a sub-class of nodes. Of course, in other embodiments what is a sub-class here can be considered a class in itself.

[0092] As noted before, the switches C_X^{1-4} 240, C_X^{5-8} 252, and C_X^{9-12} 264, are associated with the *X* fabric, F_X, shown in Figure 7. Accordingly, where such switches are associated with node *V*1

190, for example, each switch then connects to both nodes *V2* 192 and *V5* 200 according to the diagram. This same type of configuration can be used for each node of the fabric F_X such that:

$$V2_X \Rightarrow \{C_X^{13-16}(N_{13},N_{14},N_{15},N_{16}),C_X^{17-20}(N_{17},N_{18},N_{19},N_{20}),C_X^{21-24}(N_{21},N_{22},N_{23},N_{24})\}$$

$$V3_X \Rightarrow \{C_X^{25-28}(N_{25},N_{26},N_{27},N_{28}),C_X^{29-32}(N_{29},N_{30},N_{31},N_{32}),C_X^{33-36}(N_{33},N_{34},N_{35},N_{36})\}$$

$$V4_X \Rightarrow \{C_X^{37-40}(N_{37},N_{38},N_{39},N_{40}),C_X^{41-44}(N_{41},N_{42},N_{43},N_{44}),C_X^{45-48}(N_{45},N_{46},N_{47},N_{48})\}$$

$$V5_X \Rightarrow \{C_X^{49-52}(N_{49},N_{50},N_{51},N_{52}),C_X^{53-56}(N_{53},N_{54},N_{55},N_{56}),C_X^{57-60}(N_{57},N_{58},N_{59},N_{60})\}.$$

[0093] Thus, switches C_X^{1-4} 240, C_X^{5-8} 252, and C_X^{9-12} 264 are associated with the node $V1_X$ 270, switches C_X^{13-16} 272, C_X^{17-20} 274, and C_X^{21-24} 276 are associated with the node $V2_X$ 280, switches C_X^{25-28} 282, C_X^{29-32} 284, and C_X^{33-36} 286 are associated with the node $V3_X$ 290, switches C_X^{37-40} 292, C_X^{41-44} 294, and C_X^{45-48} 296 are associated with the node $V4_X$ 300, and switches C_X^{49-52} 302, C_X^{53-56} 304, and C_X^{57-60} 306 are associated with the node $V5_X$ 310.

[0094] A full implementation of the fabric F_X can then be configured as shown in Figure 12. For clarity of presentation, only the switches are shown while omitting the endpoints connected to the switches. In implementing the fabric F_X of Figure 12, inter-node connectivity is provided by 6-port routers *Ex*(v1,v2) 312, *Ex*(v2,v3) 314, *Ex*(v3,v4) 316, *Ex*(v4,v5) 320 and *Ex*(v1,v5) 322 in accordance with the ring connection:

$$F_X \Rightarrow \{E_X(V1_X,V2_X),\ E_X(V2_X,V3_X),\ E_X(V3_X,V4_X),\ E_X(V4_X,V5_X),\ E_X(V1_X,V5_X)\}.$$

Note that the subscript denotes the fabric and the argument denotes the node-to-node connections. These 6-port routers 312-322 then allow for 3-port connectivity from node to node (*e.g.*, node *V1* 270 to node *V2* 280, etc.).

[0095] In the same manner that the *X* fabric, F_X, is configured, the *Y* fabric can similarly be configured. With reference to Figure 10, the similarities to Figure 9 are evident. In particular, note that the endpoints 230-236, 242-250 and 254-262 of Figure 10 are the same as those of Figure 9. That is, each endpoint (e.g., 230) has two ports (e.g., 330), one for communication on the *X* fabric F_X and one for communication on the *Y* fabric, F_Y.

[0096] Switches C_Y^{1-4} 332, C_Y^{5-8} 334, and C_Y^{9-12} 336 of Figure 10, however, are distinct from those of Figure 9 in that they are associated with the *Y* fabric, F_Y. The following notation, therefore describes the above connections for the *Y* fabric connections of node *V1*:

[0097] $V1_Y \Rightarrow \{C_Y^{1-4}(N_1,N_2,N_3,N_4),C_Y^{5-8}(N_5,N_6,N_7,N_8),C_Y^{9-12}(N_9,N_{10},N_{11},N_{12})\}$

This same type of configuration can be used for each of nodes of the fabric F_Y such that:

$$V2_Y \rightarrow \{C_Y^{13-16}(N_{13},N_{14},N_{15},N_{16}), C_Y^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_Y^{21-24}(N_{21},N_{22},N_{23},N_{24})\}$$

$$V3_Y \rightarrow \{C_Y^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_Y^{29-32}(N_{29},N_{30},N_{31},N_{32}), C_Y^{33-36}(N_{33},N_{34},N_{35},N_{36})\}$$

$$V4_Y \rightarrow \{C_Y^{37-40}(N_{37},N_{38},N_{39},N_{40}), C_Y^{41-44}(N_{41},N_{42},N_{43},N_{44}), C_Y^{45-48}(N_{45},N_{46},N_{47},N_{48})\}$$

$$V5_Y \rightarrow \{C_Y^{49-52}(N_{49},N_{50},N_{51},N_{52}), C_Y^{53-56}(N_{53},N_{54},N_{55},N_{56}), C_Y^{57-60}(N_{57},N_{58},N_{59},N_{60})\}.$$

[0098] Thus, switches C_Y^{1-4} 332, C_Y^{5-8} 334, and C_Y^{9-12} 336 are associated with the node $V1_Y$ 340, switches C_Y^{13-16} 342, C_Y^{17-20} 344, and C_Y^{21-24} 346 are associated with the node $V2_Y$ 370, switches C_Y^{25-28} 352, C_Y^{29-32} 354, and C_Y^{33-36} 356 are associated with the node $V3_Y$ 350, switches C_Y^{37-40} 362, C_Y^{41-44} 364, and C_Y^{45-48} 366 are associated with the node $V4_Y$ 380, and switches C_Y^{49-52} 372, C_Y^{53-56} 374, and C_Y^{57-60} 376 are associated with the node $V5_Y$ 360.

[0099] With reference now to Figure 13, the similarities to Figure 12 are again evident. In Figure 13, however, the node-to-node connections are in accordance with the *Y* fabric, F_Y, configuration. Here, inter-node connectivity is provided by 6-port routers *Ey*(v1,v3) 392, *Ey*(v3,v5) 394, *Ey*(v2,v5) 396, *Ey*(v2,v4) 400 and *Ey*(v1,v4) 402 with different node-to-node connections:

$$F_Y \rightarrow \{E_Y(V1_Y,V3_Y),\ E_Y(V3_Y,V5_Y),\ E_Y(V2_Y,V5_Y),\ E_Y(V2_Y,V4_Y),\ E_Y(V1_Y,V4_Y)\}.$$

Note that here, the node-to-node connections correspond to the star configuration of Figure 6 and the reorganized ring configuration of Figure 8.

[0100] Thus, Figures 12 and 13 depict the fabrics F_X and F_Y respectively. As previously discussed, the union of the two fabrics composes the complete fabric, $F=F_X \cup F_Y$ such that

$$V1=V1_X \cup V1_Y$$

$$V1 \rightarrow \{C_X^{1-4}(N_1,N_2,N_3,N_4), C_X^{5-8}(N_5,N_6,N_7,N_8), C_X^{9-12}(N_9,N_{10},N_{11},N_{12})\} \cup$$
$$\{C_Y^{1-4}(N_1,N_2,N_3,N_4), C_Y^{5-8}(N_5,N_6,N_7,N_8), C_Y^{9-12}(N_9,N_{10},N_{11},N_{12})\}$$

$$V2=V2_X \cup V2_Y$$

$$V2 \rightarrow \{C_X^{13-16}(N_{13},N_{14},N_{15},N_{16}), C_X^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_X^{21-24}(N_{21},N_{22},N_{23},N_{24})\} \cup$$
$$\{C_Y^{13-16}(N_{13},N_{14},N_{15},N_{16}), C_Y^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_Y^{21-24}(N_{21},N_{22},N_{23},N_{24})\}$$

$$V3=V3_X \cup V3_Y$$

$$V3 \rightarrow \{C_X^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_X^{29-32}(N_{29},N_{30},N_{31},N_{32}), C_X^{33-36}(N_{33},N_{34},N_{35},N_{36})\}$$
$$\cup \{C_Y^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_Y^{29-32}(N_{29},N_{30},N_{31},N_{32}), C_Y^{33-36}(N_{33},N_{34},N_{35},N_{36})\}$$

$$V4=V4_X \cup V4_Y$$

$V4 \rightarrow \{C_X^{37-40}(N_{37},N_{38},N_{39},N_{40}),C_X^{41-44}(N_{41},N_{42},N_{43},N_{44}),C_X^{45-48}(N_{45},N_{46},N_{47},N_{48})\}$

$\cup \ \{C_Y^{37-40}(N_{37},N_{38},N_{39},N_{40}),C_Y^{41-44}(N_{41},N_{42},N_{43},N_{44}),C_Y^{45-48}(N_{45},N_{46},N_{47},N_{48})\}$

$V5 = V5_X \cup V5_Y$

$V5 \rightarrow \{C_X^{49-52}(N_{49},N_{50},N_{51},N_{52}),C_X^{53-56}(N_{53},N_{54},N_{55},N_{56}),C_X^{57-60}(N_{57},N_{58},N_{59},N_{60})\}$

$\cup \ \{C_Y^{49-52}(N_{49},N_{50},N_{51},N_{52}),C_Y^{53-56}(N_{53},N_{54},N_{55},N_{56}),C_Y^{57-60}(N_{57},N_{58},N_{59},N_{60})\}.$

[0101] To demonstrate how this may be done, Figure 14 shows the union of the configurations discussed for Figures 12 and 13. As shown in Figure 14, note that the endpoints 230-236, 242-250 and 254-262 are again the same. Here, however, the endpoints are shown with the two port connections, one for the *X* fabric and one for the *Y* fabric. Moreover, as shown in Figure 14, 4-in-2-out switches, C_X^{1-4} 240, C_X^{5-8} 252 and C_X^{9-12} 264 associated with the *X* fabric are shown along with 4-in-2-out switches, C_Y^{1-4} 332, C_Y^{5-8} 334 and C_Y^{9-12} 336 associated with the *Y* fabric. Thus we have

$F = F_X \cup F_Y$

$F \rightarrow \{E_X(V1_X,V2_X),\ E_X(V2_X,V3_X),\ E_X(V3_X,V4_X),\ E_X(V4_X,V5_X),$
$E_X(V1_X,V5_X)\} \cup$

$\{E_Y(V1_Y,V3_Y),\ E_Y(V3_Y,V5_Y),\ E_Y(V2_Y,V5_Y),\ E_Y(V2_Y,V4_Y),\ E_Y(V1_Y,V4_Y)\}.$

[0102] Thus, the complete fabric *F* is configured as shown in Figure 14. Notably, the fabric *F* allows for complete inter-node connectivity, that is, every node can directly communicate with every other node. Moreover, the fabric *F* provides for intra-class connectivity, that is, every endpoint within a class can communicate with another endpoint of the same class. More particularly, intra-sub-class connectivity is provided by the 4-in-2-out switches (e.g., 240) and inter-sub-class connectivity within the same class is provided by the 6-port routers (e.g., 312). Of course, inter-class connectivity is provided by inter-node connectivity. We therefore achieve the desirable result that every endpoint is communicatively coupled to every other endpoint.

[0103] In considering Figure 14, the concept of class-router sharing is clearly evident. For instance, the 4-in-2-out switches (e.g., 240) that provide intra-class connectivity are repeated not on a per-group basis but rather on a per-fabric basis. Thus, even though each endpoint connects to 4 total groups per the mathematical design of Figure 6, it needs only two network interface ports per the physical network topology of Figure 14, one for the X fabric and one for the Y fabric. This contrasts with the design of Figure 5, where each endpoint needed four network interface ports in order to precisely match the rank of its equivalence class. In other embodiments, whenever there is a design challenge brought about by a mismatch between the rank of a BIBD and the number of physical ports that an endpoint is constrained to use, the principle of class-router sharing may be used in accordance with the principles of the present teachings in order to overcome that design challenge. For example, with respect to Figure 14, the class router $C_X^{1\text{-}4}$ 240 is shared between the groups {V1,V2} and {V1,V5} in the X fabric and the class router $C_Y^{1\text{-}4}$ 332, between the groups {V1,V3} and {V1,V4} in the Y fabric. The embodiments that follow take advantage of class router sharing as described above.

[0104] For connecting 64 nodes using 6-port crossbar switches, the topology of Figure 14 uses only 40 switches and satisfies some highly desirable properties. For example, such a design exhibits low latency. Here, there are 3 or fewer switches on the best path between any pair of endpoints. A prior art approach, such as MINs, Clos networks and *k*-ary n-cubes, would have put 5 switches on the best path for certain node pairs. The present approach also exhibits desirable redundant connectivity. Here, there are two completely independent paths between any pair of nodes, one in the *X* fabric and one in the *Y* fabric, yet only 40 total switches are used. Prior art techniques that yield low latency and use identical fabrics for redundancy would have required 54 switches for two fabrics of a Clos network, and 48 switches for two fabrics of a 4-ary 2-cube. It should be noted, however, that the Clos network would have had a non-blocking architecture for any traffic pattern, whereas the present approach is not free of congestion and blocking. We will show below, that the present teachings can also be used to design crossbar-only interconnects, which are both non-blocking and congestion-free, as well as exhibiting lower latency than Clos networks.

[0105] It is further illustrative to consider the routing of packets within the physical network topology of Figure 14. Under normal circumstances, when all the links and switches are functional, a packet from one endpoint to another, traveling along a shortest path between those endpoints, will need to traverse an inter-class router at most once. This is so because the design of Figure 6, and indeed any design created in accordance with the principles of the present teachings, guarantees that every pair of classes is directly connected in some group. Thus, the shortest path between any pair of endpoints does not traverse the routers of more than one group. Due to this characteristic of routing in the topologies designed in accordance with the present teachings, routing domains exist within each fabric that obviate (and could preclude) the routing of packets between inter-class routers through a class router. In that sense, the present teachings specify a systematic method of creating multiple routing domains within one or more fabrics. Viewed another way, the present teachings also specify a systematic method of creating congestion domains within one or more fabrics. In particular, every group specified by the

BIBD, and translated into a portion of a physical network topology in accordance with the principles of intra-class and inter-class connectivity outlined above, corresponds to both a routing domain and a congestion domain within the fabric that contains that group.

[0106] To further illustrate other general properties of the present approach, a 2-(9,3,1) = 12 BIBD design will now be described. In this design, 12 groups are needed to connect nine elements, arranged in groups of 3, such that each pair of elements appears in only one group. Referring now to Figure 15, in the present case the nine groups of the BIBD are nine nodes of a network, *V*1 410, *V*2 412, *V*3 414, *V*4 416, *V*5 420, *V*6 422, *V*7 424, *V*8 426 and *V*9 430. These nine nodes 410-430 can be drawn as shown in Figure 15. In particular, the nine nodes 410-430 can be drawn as a grid of nodes generally arranged in three rows 432, 434 and 436 and three columns 440, 442 and 444. As part of this design, note that each node (e.g., 410) is directly connected to every other node (e.g., 430). For example, node *V*1 410 is connected to each of nodes *V*2-*V*9 412-430. Where the nine elements are nodes *V*1-*V*9 410-430, the 12 groups are therefore the node-to-node connections of the fabric, *F* 446:

$$F \Rightarrow \left\{ \begin{array}{l} \{V1, V2, V3\}, \\ \{V4, V5, V6\}, \\ \{V7, V8, V9\} \\ \{V1, V4, V7\}, \\ \{V2, V5, V8\}, \\ \{V3, V6, V9\} \\ \{V1, V5, V9\} \\ \{V2, V6, V7\} \\ \{V3, V4, V8\} \\ \{V1, V6, V8\} \\ \{V2, V4, V9\} \\ \{V3, V5, V7\} \end{array} \right\}.$$

[0107] These inter-node connections are shown in Figures 15, 16, and 17. Inter-node connection 460 connects nodes *V*1 410, *V*2 412 and *V*3 414. Inter-node connection 462 connects nodes *V*4 416, *V*5 420 and *V*6 422. Inter-node connection 464 connects nodes *V*7 424, *V*8 426 and *V*9 430. Inter-node connection 466 connects nodes *V*1 410, *V*4 416 and *V*7 424. Inter-node connection 470 connects nodes *V*2 412, *V*5 420 and *V*8 426. Inter-node connection 472 connects nodes *V*3 414, *V*6 422 and *V*9 430. Inter-node connection 480 connects nodes *V*1 410, *V*5 420 and *V*9 430. Inter-node connection 482 connects nodes *V*6 422, *V*7 424 and *V*2 412. Inter-node connection 484 connects nodes *V*8 426, *V*3 414 and *V*4 416. Inter-node connection 486 connects nodes *V*1 410, *V*6 422 and *V*8 426. Inter-node connection 490 connects nodes *V*5 420, *V*7 424 and *V*3 414. Finally, inter-node connection 492 connects nodes *V*9 430, *V*2 412 and *V*4 416. To clarify the numerous inter-node connections shown in Figure 15 as much as possible, inter-node connections in the *X* fabric are shown with straight lines and laid out as in Figure 16, inter-node connections in the *Y* fabric are shown with curved lines, and each inter-node connection contacts the circle indicating a node at a single unique point. Element numbers for inter-node connections are shown in Figures 16 and 17 but are left off in Figure 15.

[0108] This collection of inter-node connections can therefore be considered a fabric, *F*. Notably, the fabric *F* can be partitioned into two partial fabrics to be called *X* fabric, F_X 494 (Figure 16) and *Y* fabric, F_Y 496 (Figure 17). In particular we note the following partitioning of the fabric, *F*:

$$F_X \rightarrow \left\{\begin{matrix}\{V1,V2,V3\}, \\ \{V4,V5,V6\}, \\ \{V7,V8,V9\} \\ \{V1,V4,V7\}, \\ \{V2,V5,V8\}, \\ \{V3,V6,V9\}\end{matrix}\right\}$$

$$F_Y \rightarrow \left\{\begin{matrix}\{V1,V5,V9\} \\ \{V2,V6,V7\} \\ \{V3,V4,V8\} \\ \{V1,V6,V8\} \\ \{V2,V4,V9\} \\ \{V3,V5,V7\}\end{matrix}\right\}.$$

Thus, the union of F_X and F_Y provides for the fabric F=$F_X \cup F_Y$..

[0109] As shown in Figure 15, a partial grid of fabrics exists that connect nodes *V*1-*V*9 410-430 in a first horizontal and vertical pattern as shown. In one instance this configuration is called an *X* fabric, F_X 494 as shown in Figure 16. With regard to Figure 15, diagonal connections exist that serially connect nodes *V*1-*V*9 410-430 also. This diagonal pattern can be reorganized in the form of a grid with vertical and horizontal connections called a *Y* fabric, F_Y 496, with similar connections as shown in Figure 17. Accordingly, the collection of fabrics, *F* 446, as shown in Figure 15, can be redrawn as the union of two fabrics, F_X 494 and F_Y 496 (*i.e.*, F=$F_X \cup F_Y$) as shown in Figures 16 and 17, respectively.

[0110] As discussed previously, an equivalence class is a group of similarly connected nodes or endpoints which, for clarity of the discussion, we will again call endpoints while using the term "node" for the various nodes *V*1-*V*9. As shown in Figure 18, four endpoints, N_1 500, N_2 502, N_3 504 and N_4 506, are connected to a 4-in-2-out switch C_X^{1-4} 510. Similarly, four endpoints, N_5 512, N_6 514, N_7 516 and N_8 520, are connected to a 4-in-2-out switch C_X^{5-8} 522. Here, the collection of 8 endpoints 500-506 and 512-520 can be considered as an equivalence class of nodes connected to a node (for example, node *V*1 410 of Figure 16). The following notation, therefore describes the above connections for the *X* fabric connections of node *V*1 410:

$$V1_X \rightarrow \{C_X^{1-4}(N_1,N_2,N_3,N_4), C_X^{5-8}(N_5,N_6,N_7,N_8)\}$$

Each collection of four nodes 500-506 and 512-520 mentioned above is considered a sub-class of nodes. Of course, in other embodiments what is a sub-class here can be considered a class in itself.

[0111] With regard to the switches, C_X^{1-4} 510 and C_X^{5-8} 522, they are associated with the *X* fabric, F_X 494, shown in Figure 16. Accordingly, where such switches are associated with the *X* fabric connections of node *V*1 410, for example, each switch then connects to the *X* fabric connections of nodes *V*2 412 and *V*3 414, according to the diagram. This same type of configuration can be used for each node of the fabric F_X 494 such that:

$$V2_X \Rightarrow \{C_X^{9-12}(N_9, N_{10}, N_{11}, N_{12}), C_X^{13-16}(N_{13}, N_{14}, N_{15}, N_{16})\}$$

$$V3_X \Rightarrow \{C_X^{17-20}(N_{17}, N_{18}, N_{19}, N_{20}), C_X^{21-24}(N_{21}, N_{22}, N_{23}, N_{24})\}$$

$$V4_X \Rightarrow \{C_X^{25-28}(N_{25}, N_{26}, N_{27}, N_{28}), C_X^{29-32}(N_{29}, N_{30}, N_{31}, N_{32})\}$$

$$V5_X \Rightarrow \{C_X^{33-36}(N_{33}, N_{34}, N_{35}, N_{36}), C_X^{37-40}(N_{37}, N_{38}, N_{39}, N_{40})\}$$

$$V6_X \Rightarrow \{C_X^{41-44}(N_{41}, N_{42}, N_{43}, N_{44}), C_X^{45-48}(N_{45}, N_{46}, N_{47}, N_{48})\}$$

$$V7_X \Rightarrow \{C_X^{49-52}(N_{49}, N_{50}, N_{51}, N_{52}), C_X^{53-56}(N_{53}, N_{54}, N_{55}, N_{56})\}$$

$$V8_X \Rightarrow \{C_X^{57-60}(N_{57}, N_{58}, N_{59}, N_{60}), C_X^{61-64}(N_{61}, N_{62}, N_{63}, N_{64})\}$$

$$V9_X \Rightarrow \{C_X^{65-68}(N_{65}, N_{66}, N_{67}, N_{68}), C_X^{69-72}(N_{69}, N_{70}, N_{71}, N_{72})\}$$

[0112] A full implementation of the fabric F_X 494 can then be configured as shown in Figure 21. For clarity of presentation, only the endpoints of node $V1_X$ are shown, however, it should be understood that every other node is similarly connected to respective endpoints. Thus, the equivalence class of endpoints 9-12 and 13-16 (not shown) are connected to node $V2_X$ 412 by switches C_X^{9-12} 530 and C_X^{13-16} 532. The equivalence class of endpoints 17-20 and 21-24 (not shown) are connected to node $V3_X$ 414 by switches C_X^{17-20} 534 and C_X^{21-24} 536. The equivalence class of endpoints 25-28 and 29-32 (not shown) are connected to node $V4_X$ 416 by switches C_X^{25-28} 540 and C_X^{29-32} 542. The equivalence class of endpoints 33-36 and 37-40 (not shown) are connected to node $V5_X$ 420 by switches C_X^{33-36} 544 and C_X^{37-40} 546. The equivalence class of endpoints 41-44 and 45-48 (not shown) are connected to node $V6_X$ 422 by switches C_X^{41-44} 550 and C_X^{45-48} 552. The equivalence class of endpoints 49-52 and 53-56 (not shown) are connected to node $V7_X$ 424 by switches C_X^{49-52} 554 and C_X^{53-56} 556. The equivalence class of endpoints 57-60 and 61-64 (not shown) are connected to node $V8_X$ 426 by switches C_X^{57-60} 560 and C_X^{61-64} 562. Finally, the equivalence class of endpoints 65-68 and 69-72 (not shown) are connected to node $V9_X$ 430 by switches C_X^{65-68} 564 and C_X^{69-72} 566.

[0113] In implementing the fabric F_X 494 of Figure 21, inter-node connectivity is provided by 6-port routers in accordance with the grid connection:

$E_X(V1,V2,V3)$ 570,

$E_X(V4,V5,V6)$ 572,

$E_X(V7,V8,V9)$ 574,

$E_X(V1,V4,V7)$ 576,

$E_X(V2,V5,V8)$ 580, and

$E_X(V3,V6,V9)$ 582.

Note that the subscript denotes the fabric and the argument denotes the node-to-node connections. These 6-port routers 570-582 allow for 2-port connectivity to three nodes (*e.g.*, node *V*1 410 to *V*2 412 to *V*3 414, etc.).

[0114] In the same manner that the *X* fabric, F_X 494, is configured, the *Y* fabric 496 may similarly be configured. With reference to Figure 19, the similarities to Figure 18 are evident. In particular, note that the endpoints 500-506 and 512-520 of Figure 19 are the same as those of Figure 18. That is, each endpoint (e.g., 500) has two ports (e.g., 590 and 592, Figure 20), one 590 for communication on the *X* fabric F_X 494 and one 592 for communication on the *Y* fabric, F_Y 496.

[0115] Switches C_Y^{1-4} 594 and C_Y^{5-8} 596 of Figure 20, however, are distinct from those 510 and 522 of Figure 18 in that they are associated with the *Y* fabric, F_Y 496, specifically with the *Y* fabric connections of node *V*1 410. The following notation, therefore describes the above connections for the *Y* fabric connections of node *V*1:

$$V1_Y \rightarrow \{C_Y^{1-4}(N_1,N_2,N_3,N_4), C_Y^{5-8}(N_5,N_6,N_7,N_8)\}$$

This same type of configuration can be used for each of nodes of the fabric F_Y 496 such that:

$$V2_Y \rightarrow \{C_Y^{9-12}(N_9,N_{10},N_{11},N_{12}), C_Y^{13-16}(N_{13},N_{14},N_{15},N_{16})\}$$

$$V3_Y \rightarrow \{C_Y^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_Y^{21-24}(N_{21},N_{22},N_{23},N_{24})\}$$

$$V4_Y \rightarrow \{C_Y^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_Y^{29-32}(N_{29},N_{30},N_{31},N_{32})\}$$

$$V5_Y \rightarrow \{C_Y^{33-36}(N_{33},N_{34},N_{35},N_{36}), C_Y^{37-40}(N_{37},N_{38},N_{39},N_{40})\}$$

$$V6_Y \rightarrow \{C_Y^{41-44}(N_{41},N_{42},N_{43},N_{44}), C_Y^{45-48}(N_{45},N_{46},N_{47},N_{48})\}$$

$$V7_Y \rightarrow \{C_Y^{49-52}(N_{49},N_{50},N_{51},N_{52}), C_Y^{53-56}(N_{53},N_{54},N_{55},N_{56})\}$$

$V8_Y \rightarrow \{C_Y^{57-60}(N_{57}, N_{58}, N_{59}, N_{60}), C_Y^{61-64}(N_{61}, N_{62}, N_{63}, N_{64})\}$

$V9_Y \rightarrow \{C_Y^{65-68}(N_{65}, N_{66}, N_{67}, N_{68}), C_Y^{69-72}(N_{69}, N_{70}, N_{71}, N_{72})\}$.

[0116] With reference now to Figure 22, the similarities to Figure 21 are again evident. In Figure 22, however, the node-to-node connections are in accordance with the Y fabric, F_Y 496, configuration. Again, for clarity of presentation, only the endpoints of node $V1_Y$ are shown, however, it should be understood that every other node is similarly connected to respective endpoints. Thus, the equivalence class of endpoints 9-12 and 13-16 (not shown) are connected to node $V2_Y$ 412 by switches C_Y^{9-12} 600 and C_Y^{13-16} 602. The equivalence class of endpoints 17-20 and 21-24 (not shown) are connected to node $V3_Y$ 414 by switches C_Y^{17-20} 604 and C_Y^{21-24} 606. The equivalence class of endpoints 25-28 and 29-32 (not shown) are connected to node $V4_Y$ 416 by switches C_Y^{25-28} 610 and C_Y^{29-32} 612. The equivalence class of endpoints 33-36 and 37-40 (not shown) are connected to node $V5_Y$ 420 by switches C_Y^{33-36} 614 and C_Y^{37-40} 616. The equivalence class of endpoints 41-44 and 45-48 (not shown) are connected to node $V6_Y$ 422 by switches C_Y^{41-44} 620 and C_Y^{45-48} 622. The equivalence class of endpoints 49-52 and 53-56 (not shown) are connected to node $V7_Y$ 424 by switches C_Y^{49-52} 624 and C_Y^{53-56} 626. The equivalence class of endpoints 57-60 and 61-64 (not shown) are connected to node $V8_Y$ 426 by switches C_Y^{57-60} 630 and C_Y^{61-64} 632. Finally, the equivalence class of endpoints 65-68 and 69-72 (not shown) are connected to node $V9_Y$ 430 by switches C_Y^{65-68} 634 and C_Y^{69-72} 636.

[0117] Here, inter-node connectivity is provided by 6-port routers with different connections:

$E_Y(V1, V5, V9)$ 640,

$E_Y(V2, V6, V7)$ 642,

$E_Y(V3, V4, V8)$ 644,

$E_Y(V1, V6, V8)$ 646,

$E_Y(V2, V4, V9)$ 650, and

$E_Y(V3, V5, V7)$ 652.

Note that, here, the node-to-node connections produce the reorganized grid configuration of Figure 17. Thus, Figures 21 and 22 depict the fabrics F_X 494 and F_Y 496, respectively.

[0118] As previously discussed, the union of the two fabrics 494 and 496 composes the complete fabric 446, $F = F_X \cup F_Y$ such that:

$V1 = V1_X \cup V1_Y$

$V1 \rightarrow \{C_X^{1-4}(N_1, N_2, N_3, N_4), C_X^{5-8}(N_5, N_6, N_7, N_8)\} \cup$

$$\left\{C_Y^{1-4}(N_1,N_2,N_3,N_4), C_Y^{5-8}(N_5,N_6,N_7,N_8)\right\}$$

$V2 = V2_X \cup V2_Y$

$$V2 \rightarrow \left\{C_X^{9-12}(N_9,N_{10},N_{11},N_{12}), C_X^{13-16}(N_{13},N_{14},N_{15},N_{16})\right\} \cup \left\{C_Y^{9-12}(N_9,N_{10},N_{11},N_{12}), C_Y^{13-16}(N_{13},N_{14},N_{15},N_{16})\right\}$$

$V3 = V3_X \cup V3_Y$

$$V3 \rightarrow \left\{C_X^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_X^{21-24}(N_{21},N_{22},N_{23},N_{24})\right\} \cup \left\{C_Y^{17-20}(N_{17},N_{18},N_{19},N_{20}), C_Y^{21-24}(N_{21},N_{22},N_{23},N_{24})\right\}$$

$V4 = V4_X \cup V4_Y$

$$V4 \rightarrow \left\{C_X^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_X^{29-32}(N_{29},N_{30},N_{31},N_{32})\right\} \cup \left\{C_Y^{25-28}(N_{25},N_{26},N_{27},N_{28}), C_Y^{29-32}(N_{29},N_{30},N_{31},N_{32})\right\}$$

$V5 = V5_X \cup V5_Y$

$$V5 \rightarrow \left\{C_X^{33-36}(N_{33},N_{34},N_{35},N_{36}), C_X^{37-40}(N_{37},N_{38},N_{39},N_{40})\right\} \cup \left\{C_Y^{33-36}(N_{33},N_{34},N_{35},N_{36}), C_Y^{37-40}(N_{37},N_{38},N_{39},N_{40})\right\}$$

$V6 = V6_X \cup V6_Y$

$$V6 \rightarrow \left\{C_X^{41-44}(N_{41},N_{42},N_{43},N_{44}), C_X^{45-48}(N_{45},N_{46},N_{47},N_{48})\right\} \cup \left\{C_Y^{41-44}(N_{41},N_{42},N_{43},N_{44}), C_Y^{45-48}(N_{45},N_{46},N_{47},N_{48})\right\}$$

$V7 = V7_X \cup V7_Y$

$$V7 \rightarrow \left\{C_X^{49-52}(N_{49},N_{50},N_{51},N_{52}), C_X^{53-56}(N_{53},N_{54},N_{55},N_{56})\right\} \cup \left\{C_Y^{49-52}(N_{49},N_{50},N_{51},N_{52}), C_Y^{53-56}(N_{53},N_{54},N_{55},N_{56})\right\}$$

$V8 = V8_X \cup V8_Y$

$$V8 \rightarrow \left\{C_X^{57-60}(N_{57},N_{58},N_{59},N_{60}), C_X^{61-64}(N_{61},N_{62},N_{63},N_{64})\right\} \cup \left\{C_Y^{57-60}(N_{57},N_{58},N_{59},N_{60}), C_Y^{61-64}(N_{61},N_{62},N_{63},N_{64})\right\}$$

$V9 = V9_X \cup V9_Y$

$$V9 \rightarrow \left\{C_X^{65-68}(N_{65},N_{66},N_{67},N_{68}), C_X^{69-72}(N_{69},N_{70},N_{71},N_{72})\right\} \cup \left\{C_Y^{65-68}(N_{65},N_{66},N_{67},N_{68}), C_Y^{69-72}(N_{69},N_{70},N_{71},N_{72})\right\}.$$

[0119] Figure 20 shows the connectivity of node *V1* 410. The connectivity between the nodes can be inferred from superposition of Figures 21 and 22. Whereas the fabric, *F* 446, of Figure 15 was quite complex, a full implementation of the network design just described is even more complex such that a drawing is not provided. Nonetheless, the fabric *F* 446 allows for complete inter-node connectivity, in that every node can directly communicate with every other node. It should be noted, however, that in other embodiments, inter-node connectivity may be provided by way of an intermediate node, router, switch, or endpoint. The fabric *F* 446 provides for intra-class connectivity, that is, every endpoint within a class can communicate with another endpoint of the same class. More particularly, intra-sub-class connectivity is provided by the 4-in-2-out switches and inter-sub-class connectivity within the same class is provided by the 6-port routers. Of course, inter-class connectivity is provided by inter-node connectivity. We therefore achieve the desirable result that every endpoint is communicatively coupled to every other endpoint. For connecting 72 nodes using 6-port crossbar switches, the topology of Figures 15 through 22 uses a total of 48 switches. This includes 4 switches as shown in Figure 20 in order to implement each of the 9 classes, and 12 additional switches as shown in Figures 21 and 22 to implement inter-node connectivity implied by Figure 15. The present approach also satisfies some highly desirable properties. For example, such a design exhibits low latency. Here, there are 3 or fewer switches on the best path between any pair of endpoints. A prior art approach, such as MINs, Clos networks and k-ary n-cubes, would have put 5 switches on the best path for certain node pairs. Considering that the delay of many computer operations is proportional to the round-trip time through the network, the present teachings result in 40% savings for certain latency-critical operations. The present approach also exhibits desirable redundant connectivity. Here, there are two completely independent paths between any pair of nodes, one in the *X* fabric 494 and one in the *Y* fabric 496. The 48 total switches used here are an improvement over the 54 switches for two fabrics of a Clos network. It should be noted, however, that the Clos network would have had a non-blocking architecture for any traffic pattern, whereas the present approach is not free of congestion and blocking. We will show below, that the present teachings can also be used to design crossbar-only interconnects, which are both non-blocking and congestion-free, as well as exhibiting lower latency than Clos networks.

[0120] It should also be noted that the two physical fabrics 494 and 496 shown in Figures 21 and 22 are indeed 6-ary 2-cubes. Instead of specifying two identical fabrics, as prior art would have done, the present teachings specify two asymmetric fabrics, thereby reducing latency as much as 40%. In that sense, the present teachings also provide a formal method for designing asymmetric fabric interconnects, containing two complete but non-identical fabrics.

[0121] In the particular case just described, a 72-endpoint (or 72-node) topology has been implemented using 6-port crossbar switches. Many variations exist for this particular design and for the more general designs of the present approach. Importantly, many of the results of the above-described example can be generalized for broader applicability. For example, the number of endpoints in a class may be varied to create larger or smaller classes; similarly for the sub-classes. Moreover, the configuration of the above described example can be changed to accommodate various types of available hardware. For example, a 4-in-2-out switch was described. Where a different type of switch is available, such as a 3-in-3-out switch, the network design can be modified; similarly, for the described 6-port router. Indeed the design can be optimized to accommodate available hardware.

[0122] The above-described example can further be generalized where any inter-node, inter-class, or inter-sub-class connection can be implemented as a network design in accordance with the principles herein. In this way, a large fabric can be a hierarchical collection of various fabrics of different size.

[0123] In the two examples described above, drawings were provided that illustrated node-to-node connections, partitioning, as well as partial or complete fabrics. For larger designs, however, drawings become of limited value because of the unwieldy complexity of such network designs. Accordingly, a third example describing a 2-(13, 4, 1) = 13 BIBD will be described based on an understanding of the underlying mathematical concepts of BIBDs, but without a graphical representation. In this design, 13 groups are needed to connect 13 elements, arranged in groups of 4, such that a pair of elements appears in each group. In the present case the 13 groups of the BIBD are 13 nodes of a network. Although not necessary, each node is directly connected to every other node. For example, node $V1$ is connected to each of nodes $V2$-$V13$. Where the 13 elements are nodes $V1$-$V13$, the 13 groups are therefore the node-to-node connections are therefore the fabric, F:

$$F \rightarrow \left\{ \begin{array}{l} F_1 = \{V1, V2, V4, V10\}, \\ F_2 = \{V1, V3, V9, V13\}, \\ F_3 = \{V1, V5, V6, V8\} \\ F_4 = \{V1, V7, V11, V12\}, \\ F_5 = \{V2, V3, V5, V11\}, \\ F_6 = \{V3, V5, V7, V9\} \\ F_7 = \{V2, V8, V12, V13\} \\ F_8 = \{V3, V4, V6, V12\} \\ F_9 = \{V3, V7, V8, V10\} \\ F_{10} = \{V4, V5, V7, V13\} \\ F_{11} = \{V4, V8, V9, V11\} \\ F_{12} = \{V5, V9, V10, V12\} \\ F_{13} = \{V6, V10, V11, V13\} \end{array} \right\}$$

[0124] This collection of inter-node connections can therefore be considered a fabric, F. Whereas the previous two examples were described with further partitioning into X and Y

fabrics, the present embodiment being described will use no further partitioning, but will use the natural partitioning of the BIBD design such that, in effect, four fabric interfaces per node and thirteen separate fabrics will be used.

[0125] We now turn to what has been referenced above as classes of nodes with five endpoints in a class. Using a 5-in-1-out router, the total of 65 endpoints are organized into 13 classes. The five endpoints, N_1, N_2, N_3, N_4, and N_5, of the first class are connected to four separate 5-in-1-out switches C_1^{1-5} (note that we use similar notation here as before, however, instead of using an X subscript to denote the X fabric, a number, here "1," is used), C_2^{1-5}, C_3^{1-5}, and C_4^{1-5}. Together, the four switches will allow the first class to connect into the first four fabrics, just as node $V1$ of the BIBD participates in the first four groups. The remaining twelve classes are similarly structured. The five endpoints, N_6, N_7,N_8, N_9, and N_{10}, of the second class are connected to four 5-in-1-out switches C_1^{6-10}, C_2^{6-10}, C_3^{6-10}, and C_4^{6-10}. This arrangement is repeated thirteen times, until we have the five endpoints, N_{61}, N_{62}, N_{63}, N_{64}, and N_{65}, connected to four 5-in-1-out switches C_1^{61-65}, C_2^{61-65}, C_3^{61-65}, and C_4^{61-65}. The following notation, therefore describes the above connections for the various fabric connections of the nodes $V1$ through $V13$:

$$V1 \rightarrow \left\{\begin{matrix} C_1^{1-5}(N_1,N_2,N_3,N_4,N_5), \\ C_2^{1-5}(N_1,N_2,N_3,N_4,N_5), \\ C_3^{1-5}(N_1,N_2,N_3,N_4,N_5), \\ C_4^{1-5}(N_1,N_2,N_3,N_4,N_5) \end{matrix}\right\}$$

$$V2 \rightarrow \left\{\begin{matrix} C_1^{6-10}(N_6,N_7,N_8,N_9,N_{10}), \\ C_2^{6-10}(N_6,N_7,N_8,N_9,N_{10}), \\ C_3^{6-10}(N_6,N_7,N_8,N_9,N_{10}), \\ C_4^{6-10}(N_6,N_7,N_8,N_9,N_{10}) \end{matrix}\right\}$$

$$V3 \rightarrow \left\{\begin{matrix} C_1^{11-15}(N_{11},N_{12},N_{13},N_{14},N_{15}), \\ C_2^{11-15}(N_{11},N_{12},N_{13},N_{14},N_{15}), \\ C_3^{11-15}(N_{11},N_{12},N_{13},N_{14},N_{15}), \\ C_4^{11-15}(N_{11},N_{12},N_{13},N_{14},N_{15}) \end{matrix}\right\}$$

$$V4 \rightarrow \left\{\begin{matrix} C_1^{16-20}(N_{16},N_{17},N_{18},N_{19},N_{20}), \\ C_2^{16-20}(N_{16},N_{17},N_{18},N_{19},N_{20}), \\ C_3^{16-20}(N_{16},N_{17},N_{18},N_{19},N_{20}), \\ C_4^{16-20}(N_{16},N_{17},N_{18},N_{19},N_{20}) \end{matrix}\right\}$$

$$V5 \rightarrow \begin{Bmatrix} C_1^{21-25}(N_{21}, N_{22}, N_{23}, N_{24}, N_{25}), \\ C_2^{21-25}(N_{21}, N_{22}, N_{23}, N_{24}, N_{25}), \\ C_3^{21-25}(N_{21}, N_{22}, N_{23}, N_{24}, N_{25}), \\ C_4^{21-25}(N_{21}, N_{22}, N_{23}, N_{24}, N_{25}) \end{Bmatrix}$$

$$V6 \rightarrow \begin{Bmatrix} C_1^{26-30}(N_{26}, N_{27}, N_{28}, N_{29}, N_{30}), \\ C_2^{26-30}(N_{26}, N_{27}, N_{28}, N_{29}, N_{30}), \\ C_3^{26-30}(N_{26}, N_{27}, N_{28}, N_{29}, N_{30}), \\ C_4^{26-30}(N_{26}, N_{27}, N_{28}, N_{29}, N_{30}) \end{Bmatrix}$$

$$V7 \rightarrow \begin{Bmatrix} C_1^{31-35}(N_{31}, N_{32}, N_{33}, N_{34}, N_{35}), \\ C_2^{31-35}(N_{31}, N_{32}, N_{33}, N_{34}, N_{35}), \\ C_3^{31-35}(N_{31}, N_{32}, N_{33}, N_{34}, N_{35}), \\ C_4^{31-35}(N_{31}, N_{32}, N_{33}, N_{34}, N_{35}) \end{Bmatrix}$$

$$V8 \rightarrow \begin{Bmatrix} C_1^{36-40}(N_{36}, N_{37}, N_{38}, N_{39}, N_{40}), \\ C_2^{36-40}(N_{36}, N_{37}, N_{38}, N_{39}, N_{40}), \\ C_3^{36-40}(N_{36}, N_{37}, N_{38}, N_{39}, N_{40}), \\ C_4^{36-40}(N_{36}, N_{37}, N_{38}, N_{39}, N_{40}) \end{Bmatrix}$$

$$V9 \rightarrow \begin{Bmatrix} C_1^{41-45}(N_{41}, N_{42}, N_{43}, N_{44}, N_{45}), \\ C_2^{41-45}(N_{41}, N_{42}, N_{43}, N_{44}, N_{45}), \\ C_3^{41-45}(N_{41}, N_{42}, N_{43}, N_{44}, N_{45}), \\ C_4^{41-45}(N_{41}, N_{42}, N_{43}, N_{44}, N_{45}) \end{Bmatrix}$$

$$V10 \rightarrow \begin{Bmatrix} C_1^{46-50}(N_{46}, N_{47}, N_{48}, N_{49}, N_{50}), \\ C_2^{46-50}(N_{46}, N_{47}, N_{48}, N_{49}, N_{50}), \\ C_3^{46-50}(N_{46}, N_{47}, N_{48}, N_{49}, N_{50}), \\ C_4^{46-50}(N_{46}, N_{47}, N_{48}, N_{49}, N_{50}) \end{Bmatrix}$$

$$V11 \rightarrow \begin{Bmatrix} C_1^{51-55}(N_{51}, N_{52}, N_{53}, N_{54}, N_{55}), \\ C_2^{51-55}(N_{51}, N_{52}, N_{53}, N_{54}, N_{55}), \\ C_3^{51-55}(N_{51}, N_{52}, N_{53}, N_{54}, N_{55}), \\ C_4^{51-55}(N_{51}, N_{52}, N_{53}, N_{54}, N_{55}) \end{Bmatrix}$$

$$V12 \rightarrow \begin{Bmatrix} C_1^{56-60}(N_{56}, N_{57}, N_{58}, N_{59}, N_{60}), \\ C_2^{56-60}(N_{56}, N_{57}, N_{58}, N_{59}, N_{60}), \\ C_3^{56-60}(N_{56}, N_{57}, N_{58}, N_{59}, N_{60}), \\ C_4^{56-60}(N_{56}, N_{57}, N_{58}, N_{59}, N_{60}) \end{Bmatrix}$$

$$V13 \rightarrow \left\{\begin{matrix} C_1^{61-65}(N_{61},N_{62},N_{63},N_{64},N_{65}), \\ C_2^{61-65}(N_{61},N_{62},N_{63},N_{64},N_{65}), \\ C_3^{61-65}(N_{61},N_{62},N_{63},N_{64},N_{65}), \\ C_4^{61-65}(N_{61},N_{62},N_{63},N_{64},N_{65}) \end{matrix}\right\}.$$

[0126] In implementing the thirteen fabrics F_1-F_{13}, inter-node connectivity is provided by 6-port routers E_1 to E_{13} in accordance with the connections:

$$F_1 \rightarrow E_1\left(C_1^{1-5}, C_1^{6-10}, C_1^{16-20}, C_1^{46-50}\right)$$

corresponding to {$V1$, $V2$, $V4$, $V10$} discussed above. Similarly, we have

$$F_2 \rightarrow E_2\left(C_2^{1-5}, C_1^{11-15}, C_1^{41-45}, C_1^{61-65}\right)$$

$$F_3 \rightarrow E_3\left(C_3^{1-5}, C_1^{21-25}, C_1^{26-30}, C_1^{36-40}\right)$$

$$F_4 \rightarrow E_4\left(C_4^{1-5}, C_1^{31-35}, C_1^{51-55}, C_1^{56-60}\right)$$

$$F_5 \rightarrow E_5\left(C_2^{6-10}, C_2^{11-15}, C_2^{21-25}, C_2^{51-55}\right)$$

$$F_6 \rightarrow E_6\left(C_3^{6-10}, C_2^{26-30}, C_2^{31-35}, C_2^{41-45}\right)$$

$$F_7 \rightarrow E_7\left(C_4^{6-10}, C_2^{36-40}, C_2^{56-60}, C_2^{61-65}\right)$$

$$F_8 \rightarrow E_8\left(C_4^{11-15}, C_2^{16-20}, C_3^{26-30}, C_3^{56-60}\right)$$

$$F_9 \rightarrow E_9\left(C_5^{11-15}, C_3^{31-35}, C_3^{36-40}, C_2^{46-50}\right)$$

$$F_{10} \rightarrow E_{10}\left(C_3^{16-20}, C_3^{21-25}, C_4^{31-35}, C_3^{61-65}\right)$$

$$F_{11} \rightarrow E_{11}\left(C_4^{16-20}, C_4^{36-40}, C_3^{41-45}, C_3^{51-55}\right)$$

$$F_{12} \rightarrow E_{12}\left(C_4^{21-25}, C_4^{41-45}, C_3^{46-50}, C_4^{56-60}\right)$$

$$F_{13} \rightarrow E_{13}\left(C_4^{26-30}, C_4^{46-50}, C_4^{51-55}, C_4^{61-65}\right).$$

Note that where 6-port routers are used here, only four ports are used in this implementation. Further note that the collection of these 13 partial fabrics, F_1-F_{13}, makes up the complete fabric, F. No attempt is made to depict any one of these fabrics, much less the complete fabric, because of its complexity. Upon understanding the first two embodiments with the accompanying drawings, one of skill in the art will understand that this third embodiment is simply an extension of the previous teachings that can be implemented in a real-world design.

[0127] Notably, the fabric F allows for complete inter-node connectivity, where every node can directly communicate with every other node. It should be noted, however, that in other embodiments, inter-node connectivity may be provided by way of an intermediate node, router, switch, or endpoint. The fabric F provides for intra-class connectivity where every endpoint within a class can communicate with another endpoint of the same class. More particularly, intra-sub-class connectivity is provided by the 5-in-1-out switches and inter-sub-class connectivity within the same class is provided by the 6-port routers. Of course, inter-class connectivity is provided by inter-node connectivity. We therefore achieve the desirable result that every endpoint is communicatively coupled to every other endpoint.

[0128] In the particular case just described, a 65-endpoint (or 13-node) topology has been implemented using 6-port crossbar switches. Many variations exist for this particular design and for the more general designs of the present disclosure. Importantly, many of the results of the above-described example can be generalized for broader applicability. For example, the number of endpoints in a class can be varied to create larger or smaller classes; similarly for the sub-classes. Moreover, the configuration of the above-described example can be changed to accommodate various types of available hardware. For example, a 4-in-2-out switch was described. Where a different type of switch is available, such as a 3-in-3-out switch, the network design can be modified; similarly, for the described 6-port router. Indeed the design can be optimized to accommodate available hardware.

[0129] The above-described example can further be generalized wherein any inter-node, inter-class, or inter-sub-class connection can be implemented as a network design. In this way, a large fabric can be a hierarchical collection of various fabrics of different size.

[0130] In yet another example, a 2-(9, 3, 1) = 12 BIBD is shown in Figure 23. In this design, 12 groups are needed to connect 9 elements, arranged in groups of 3, such that a pair of elements appears in each group. As shown in Figure 23, the 9 groups of the BIBD are 9 classes of similarly connected nodes *V*1 670, *V*2 672, *V*3 674, *V*4 676, *V*5 680, *V*6 682, *V*7 684, *V*8 686, and *V*9 690 of a network. Although not necessary, each class of nodes is directly connected to every other node by way of one crossbar switch (equivalently a router) of twelve crossbar switches *R*1 692, and connected to *R*2 694, *R*3 696, *R*4 700, *R*5 702, *R*6 704, *R*7 706, *R*8 710, *R*9 712, *R*10 714, *R*11 716 and *R*12 720. For example, node *V*1 670 is connected to nodes *V*2 672 and *V*3 674 by crossbar switch *R*1 692, nodes *V*4 676 and *V*7 684 by crossbar *R*7 706; other connections are as shown in Figure 23. The fabric, *F* 722, can therefore be described as:

$$F \rightarrow \left\{ \begin{array}{l} \{V1,V2,V3\}, \\ \{V3,V4,V8\}, \\ \{V4,V5,V6\}, \\ \{V2,V6,V7\}, \\ \{V1,V5,V9\}, \\ \{V7,V8,V9\}, \\ \{V1,V4,V7\}, \\ \{V3,V5,V7\}, \\ \{V2,V5,V8\}, \\ \{V1,V6,V8\}, \\ \{V3,V6,V9\}, \\ \{V2,V4,V9\} \end{array} \right\}.$$

[0131] In proceeding to develop a physical design, the various crossbar switches, $R1$-$R12$ 692-720, will be implemented as 12-port crossbar switches. With this physical implementation as a consideration, the various nodes $V1$-$V9$ 670-690 are provided as groups of similarly configured nodes or classes of nodes. More particularly, each class is implemented as four endpoints:

$$V1 \rightarrow \{N1,N2,N3,N4\}$$

$$V2 \rightarrow \{N5,N6,N7,N8\}$$

$$V3 \rightarrow \{N9,N10,N11,N12\}$$

$$V4 \rightarrow \{N13,N14,N15,N16\}$$

$$V5 \rightarrow \{N17,N18,N19,N20\}$$

$$V6 \rightarrow \{N21,N22,N23,N24\}$$

$$V7 \rightarrow \{N25,N26,N27,N28\}$$

$$V8 \rightarrow \{N29,N30,N31,N32\}$$

$$V9 \rightarrow \{N33,N34,N35,N36\}.$$

As further shown in Figure 24, as well as Figure 23, each endpoint 722 connects to four crossbar switches. For example, endpoint $N36$ 724 connects to crossbar switches $R5$ 702, $R6$ 704, $R11$ 716 and $R12$ 720. To do this, the various endpoints 722 may utilize dual NICs, where each NIC has two ports, for a total of four ports per endpoint. Where previous examples within the present disclosure described the use of class routers, no class routers are used in the present embodiment because the class router functions are performed by the various NICs of the four endpoints of a class. Accordingly, no further partitioning is necessary.

[0132] As noted, 12-port crossbar switches $R1$-$R12$ 692-720 are used such that each crossbar switch (e.g., $R1$ 692) can connect three nodes (e.g., $V1$ 670, $V2$ 672 and $V3$ 674). By similarly connecting each endpoint of a class according to the fabric, F 722, described above, the network design of Figure 24 is obtained. Notably, the fabric F 722 allows for complete inter-node and inter-endpoint connectivity, where every node (e.g., 670) can directly communicate with every other node (e.g., 672-690) and every endpoint (e.g., 724) can directly communicate with every other endpoint 722. Whereas here, node-to-node and in turn endpoint-to-endpoint connectivity is provided by crossbar switches, other embodiments are possible that provide inter-node or inter-endpoint connectivity by way of an intermediate node, router, switch, or endpoint. In the present embodiment, it should be noted that the crossbar switches also provide for inter-class (and intra-sub-class) connectivity to achieve the desirable result that every endpoint is communicatively coupled to every other endpoint.

[0133] Many variations exist for this particular design and for the more general designs of the present disclosure. Importantly, many of the results of the above-described example may be generalized for broader applicability. For example, the number of endpoints in a class may be varied to create larger or smaller classes; similarly for the sub-classes. Moreover, the configuration of the above-described example may be changed to accommodate various types of available hardware or desired fault tolerance. Figure 25 provides an example of a fault-tolerant design that is a variation of the 2-(9, 3, 1) = 12 design just described. In Figure 25, note that the nine nodes V1-V9 730 contain three endpoints each 732 as

$V1 \rightarrow \{N1, N2, N3\}$

$V2 \rightarrow \{N4, N5, N6\}$

$V3 \rightarrow \{N7, N8, N9\}$

$V4 \rightarrow \{N10, N11, N12\}$

$V5 \rightarrow \{N13, N14, N15\}$

$V6 \rightarrow \{N16, N17, N18\}$

$V7 \rightarrow \{N19, N20, N21\}$

$V8 \rightarrow \{N22, N23, N24\}$

$V9 \rightarrow \{N25, N26, N27\}$.

Further note that crossbar switches 734 connect the various nodes 730 and endpoints 732 in a manner similar to that of Figure 24 to provide complete inter-node, inter-class, intra-node, and intra-class connectivity as before. In Figure 25, however, note that crossbar-to-crossbar connections are provided by two ports (e.g., 736) of each (e.g., 740) of the 12-port crossbar switches 734. While this physical implementation has fewer endpoints 732 than the implementation of Figure 24, it advantageously provides for a fault-tolerant implementation by providing redundant, although longer, paths between nodes 730 and endpoints 732. In the design of Figure 25, three of the four inter-node (as opposed to intra-node) paths yields a two-hop contended connection, and one path provides a one-hop contention-free connection, where the latter is the preferred path, but the former provide fault tolerant paths upon a crossbar failure.

[0134] The above-described example can further be generalized wherein any inter-node, inter-class, or inter-sub-class connection can be implemented as a network design. In this way, a large fabric can be a hierarchical collection of various fabrics of different size.

[0135] In an embodiment, the present teachings are practiced on a computer system 750 as shown in Figure 26. Referring to Figure 26, an exemplary computer system 750 (*e.g.*, personal computer, workstation, mainframe, *etc.*) upon which the present teachings may be practiced is shown. When configured to practice the present teachings, system 750 becomes a computer aided design (CAD) tool suitable for assisting in designing interconnect systems in large and small scale applications. Computer system 750 is configured with a data bus 752 that communicatively couples various components. As shown in Figure 26, processor 754 is coupled to bus 752 for processing information and instructions. A computer readable volatile memory such as RAM 756 is also coupled to bus 752 for storing information and instructions for the processor 754. Moreover, computer readable read only memory (ROM) 760 is also coupled to bus 752 for storing static information and instructions for processor 754. A data storage device 762 such as a magnetic or optical disk media is also coupled to bus 752. Data storage device 762 is used for storing large amounts of information and instructions. An alphanumeric input device 764, including alphanumeric and function keys, is coupled to bus 752 for communicating information and command selections to the processor 754. A cursor control device 766 such as a mouse is coupled to bus 752 for communicating user input information and command selections to the central processor 754. Input/output communications port 770 is coupled to bus 752 for communicating with a network, other computers, or other processors, for example. Display 772 is coupled to bus 752 for displaying information to a computer user. Display device 772 may be a liquid crystal device, cathode ray tube, or other display device suitable for creating graphic images and alphanumeric characters recognizable by the user. The alphanumeric input 764 and cursor control device 766 allow the computer user to dynamically signal the two-dimensional movement of a visible symbol (pointer) on display 772.

[0136] While various embodiments and advantages have been described, it will be recognized that a number of variations will be readily apparent. For example, in implementing equivalence classes, designs can be scaled to implement networks of many sizes. Moreover, the present teachings can be used to create routing domains or virtual SANs within a larger physical fabric. Thus, the present teachings may be widely applied consistent with the foregoing disclosure and the claims which follow.